Web History Visualisation for Forensic Investigations

Sarah Lowman[1] and Ian Ferguson[2]

[1] sarah@lowmanio.co.uk
[2] University of Abertay
I.Ferguson@abertay.ac.uk

Abstract. Current tools for analysing web history often produce large amounts of data. This data is usually presented in a tabular format, which makes it difficult for forensic investigators to spot patterns and trends. Visualisations can vastly improve the time it takes to inspect and analyse large amounts of data. This paper introduces Webscavator, a new tool that provides a suite of visualisations for web history analysis. We present the results from a usability study that shows our visualisations perform well when compared to a traditional web forensics tool, Net Analysis.

Keywords: web history, visulisation, HCI, user testing, digital forensics, Webscavator, Net Analysis

1 Introduction

Web history analysis is often an important part of a digital investigation. Critical evidence can sometimes lurk in a suspect's online searches, web-based email and web history. In cases that involve crimes predominately carried out using web browsers, such as child pornography and credit-card fraud, web history analysis alone can implicate or exonerate a suspect. In other cases, web history may provide an alibi, a general profile of a suspect or indicate behavioural patterns. Web history can also provide crucial evidence in crimes unrelated to computers, such as in the case of Neil Entwhistle, who was convicted of murdering his wife and baby daughter after forensic investigators found he had performed a Google search for "how to kill with a knife" [1].

Today's digital forensic investigator has "hundreds of specific and unique application software packages and hardware devices that could qualify as cyber forensic tools... hundreds of utilities available for the same task" [2]. Usually, the basic requirement for a computer forensics tool is to convert specific files into a human readable format for analysis by a forensic investigator. This analysis is often difficult and time-consuming and involves trawling through large amounts of text. Good visual interfaces and visualisations can vastly improve the time it takes to analyse data [3–5]. They can help users gain an overview of data, spot patterns and anomalies and so reduce errors and tedium.

In this paper, we introduce Webscavator, a new tool that provides a suite of visualisations for web history data [3]. Webscavator was evaluated against a popular non-visual tool, Net Analysis[3]. The evaluation was a small-scale usability study that asked three professional investigators to complete a series of analysis tasks using both Webscavator and Net Analysis. The results indicate that Webscavator improved the accuracy, speed and confidence of the participants' answers. In addition, questionnaire results suggest that the provided visualisations would be useful in forensic investigations, and would increase efficiency, reduce errors, focus attention and help non-technical people to understand an investigator's conclusions.

In this paper, we first describe the background and related work. We then give a description of Webscavator and its various visualisations. We then present the usability study results. Finally, we reach the conclusion that well-chosen visualisations can be a useful tool for investigators analysing web history.

2 Background and Related Work

2.1 Visualisation

Visualisations can act as a temporary storage area for human cognitive processes, reducing the amount of information that must be held in working memory. This allows the brain to hold and process more information simultaneously, aiding analysis [4]. Graphical representations of mathematical problems, for example, can often increase efficiency in problem-solving by allowing difficult mathematical inferences to be substituted for easier perceptual inferences [5].

The human visual system is able to perceive graphical information such as pictures, videos and charts in parallel, but text only sequentially [6]. Images are interpreted much faster than textual descriptions as the brain processes visual input much earlier than textual input [7]. This means that visualisations can help extensively when spotting patterns and correlations across a large dataset.

Visualisations are only effective when the right kind of pictorial representation is chosen and can be manipulated to show useful information. Shneiderman's visual information seeking mantra: "overview first, zoom and filter, then details-on-demand" [8] describes what a good visualisation should provide. His advice is based on the fact that usually the goal in information retrieval is to find a narrow subset of items that match a particular query, or to develop an understanding of unexpected patterns within a set of data [9].

2.2 Related Work

So far there has been little work on visualisation specifically for web history. There are, however, generic timeline-based visualisation tools that can accept and process web log files. One of these is CyberForensic TimeLab (CFTL), a timeline-based forensic tool which finds and plots all forensic data based on

[3] http://www.digital-detective.co.uk/netanalysis.asp

timestamps [10]. As time is a ubiquitous attribute of forensic data, Olsson and Boldt believe timelines are an effective way of organising and browsing evidence. A user study that compared CFTL with Forensics Tool Kit (FTK)[4] showed striking improvements in the time taken to answer questions when users used CFTL.

Two visual representations for file information were developed by Teerlink [11]. One uses coloured square blocks to represent files in a directory, with the intensity of the colour indicating an attribute such as file type or size. This makes anomalies visually obvious. The other is a tree-map in which files and directories are shown hierarchically as coloured blocks. The size of each block indicates the space occupied in the containing directory. This tree-map makes it easier to see what files are in what folders and reduces the time it takes to locate deeply nested large files. User study results show the effectiveness of these visualisations [7].

Another file visualisation was proposed by Stamps et al [12]. Instead of concentrating on the location and attributes of the file, this focuses on the words found in file contents. A pre-design task analysis showed the difficulty and importance of searching for words related to those already found. To make this task easier, the visualisation shows the importance of particular words and their relation to other information.

Krasser et al have produced a visualisation for network activity [13]. Their aim was to make it possible for the investigator to see what was happening on the network at a glance, but to also provide more details when required. The visualisation comprises of lines from the left (IP addresses) travelling towards the right (Port numbers) with the intensity and colour of the lines representing the age and protocol used.

Tudumi is a tool that uses log file visualisation to help with monitoring and auditing user behaviours on a server [14]. It focuses on three activities: access to the server from other computers, logins to the server and identity changes from one user to another. Tudumi works by applying rules to server log files to extract information, which is then visualised.

3 Webscavator

3.1 Implementation

Webscavator is a web application written in a mixture of Python and JavaScript. For storing data, it uses SQLite[5], a lightweight open-source RDBMS. As Python's standard library includes both a built-in webserver and SQLite, deployment on a local machine is easy. By default, Webscavator will listen on a local port so that no other networked devices can gain access.

Several third-party Python libraries are used. These can be installed with the widely-used setuptools package management system[6]. The third-party JavaScript

[4] http://accessdata.com/products/forensic-investigation/ftk
[5] http://sqlite.org
[6] http://pypi.python.org/pypi/setuptools

libraries used for building the visualisations are distributed as part of Webscavator.

We built a web application rather than a stand-alone application both because of the ubiquity of modern web browsers and the excellent UI libraries available for JavaScript. People are also increasingly familiar how to use web applications. In particular, forensic investigators may already have used Autopsy[7], a web-based graphical interface for SleuthKit, which is run and used in a similar way to Webscavator.

3.2 Features

For input, Webscavator accepts CSV and XML files from programs such as Net Analysis and Web Historian[8]. This alleviates Webscavator from the complexities of directly parsing web history files. Additionally, if necessary, forensic investigators can use these tools to pre-process the data to be visualised. It is easy to extend Webscavator to handle additional CSV, XML or other formats.

Webscavator has five tabs, each showing a different visualisation. *Filters* can be applied to some of the tabs. Filters allow the user to highlight or remove web history entries from the visualisations by matching various attributes. They can be used to narrow down results, highlight particular features of interest or reduce clutter due to irrelevant entries. There are several filters provided by default, including *Google searches*, *local files*, *work hours*, *advert URLs* and *social networking URLs*. The filter bar, shown in Figure 1, is always present at the top of the page, and a filter is activated by clicking on a *highlight* or *remove* radio button. Highlighted filters have their filter name shown in green, and removal filters in red to make the current filter settings visually obvious.

Fig. 1. Screenshot of the filter bar with 'Social Networking URLs' highlighted and 'News URLs' removed

There is an *add filter* form, shown in Figure 2, that allows custom filters to be created. Users can choose an attribute, and filter based on whether it matches a given value exactly, partially or not at all; matches a regular expression; is greater than or less than a value (for date/time attributes); or is part of a given list. To specify these lists, the user can create text files in a special 'lists folder', with each line in the file treated as a list entry. There are two lists provided by

[7] http://www.sleuthkit.org/autopsy/
[8] http://www.mandiant.com/products/free_software/web_historian

default, one containing the domain names for social networking sites and the other containing the domain names of advertisers. These are used in the *Social Networking URLs* and *Advert URLs* filters.

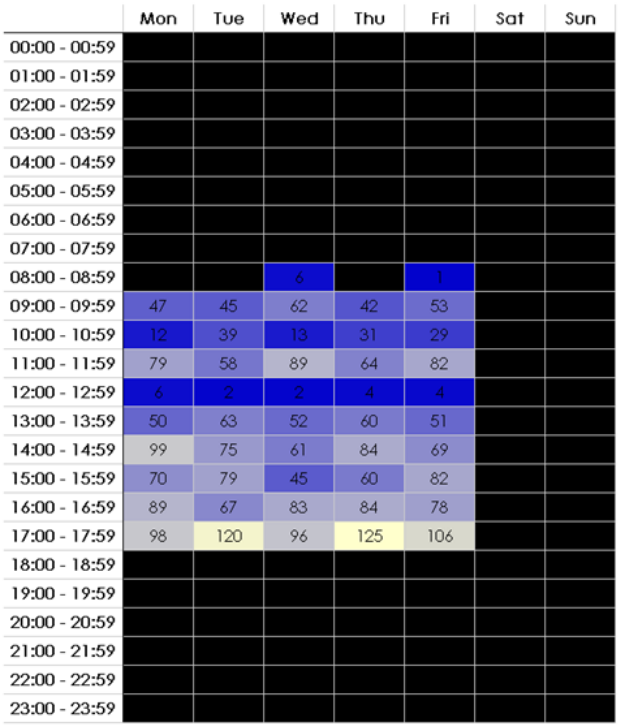
Fig. 2. Screenshot of the popup dialogue used to add a custom filter.

On the first tab, labelled 'Overview', there is a heat-map showing the number of accesses that occur for each hour of each day of the week. In a heat-map, different values are represented as different colours. It is the most widely used visualisation in biology [15]. In this case, the higher the value, the lighter the colour - small values are dark blue and high values are light blue. Webscavator's heat-map can highlight patterns in web browser usage times. Figure 3 shows the heat-map for some example data. From the colours, it is easy to see that this person used the internet between 9am and 6pm on weekdays and usage dropped significantly from 12pm to 1pm, possibly due to a lunch break. Any unusual access times, such as Sunday at 6am, would instantly stand out due to a colour change.

The second tab, 'Timeline', displays the main visualisation: a timeline, shown in Figure 4. The timeline has the day along the x-axis and the time along the y-axis. Each point on the graph is a web history entry. Weekends have a background colour of light grey. Double-clicking on part of the graph causes it to zoom centred on that part. The units of the y-axis are scaled automatically depending on zoom level. By clicking and dragging, the graph can be panned.

There is a second, smaller timeline that provides an overview. This maintains its perspective when the main timeline is zoomed or filtered, allowing the user to keep a sense of context. When the main graph is zoomed in, the overview graph shows a blue rectangle indicating which part of the data the main graph is showing, as can be seen in Figure 5. By dragging a rectangle on the overview graph, the main graph can be quickly set to show an area of interest. Hovering over a particular point will display a pop-up giving the time, date and URI of the corresponding history entry. Clicking on a point will display more detailed information about that entry underneath the plot. Entries can be excluded or

Fig. 3. Heatmap visualisation showing total number of accesses in particular hours.



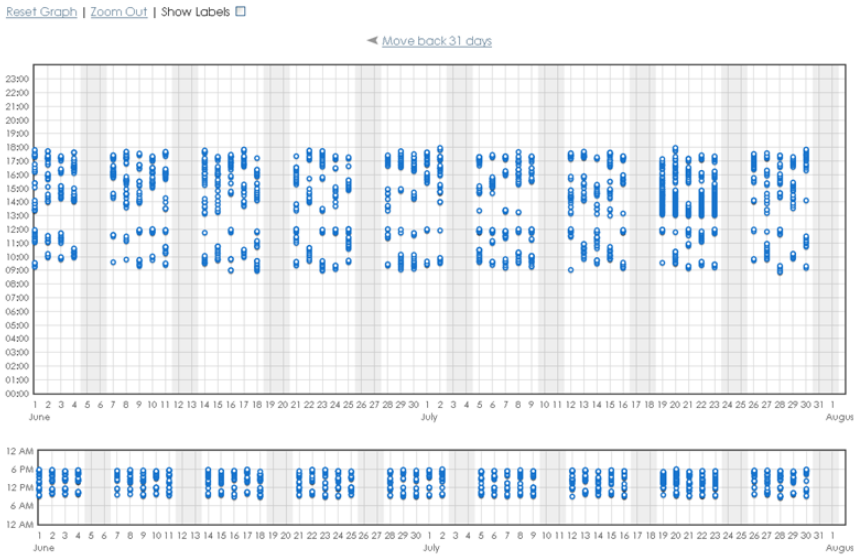
	Mon	Tue	Wed	Thu	Fri	Sat	Sun
00:00 - 00:59							
01:00 - 01:59							
02:00 - 02:59							
03:00 - 03:59							
04:00 - 04:59							
05:00 - 05:59							
06:00 - 06:59							
07:00 - 07:59							
08:00 - 08:59			6		1		
09:00 - 09:59	47	45	62	42	53		
10:00 - 10:59	12	39	13	31	29		
11:00 - 11:59	79	58	89	64	82		
12:00 - 12:59	6	2	2	4	4		
13:00 - 13:59	50	63	52	60	51		
14:00 - 14:59	99	75	61	84	69		
15:00 - 15:59	70	79	45	60	82		
16:00 - 16:59	89	67	83	84	78		
17:00 - 17:59	98	120	96	125	106		
18:00 - 18:59							
19:00 - 19:59							
20:00 - 20:59							
21:00 - 21:59							
22:00 - 22:59							
23:00 - 23:59							

highlighted using filters. Excluded entries turn a faded grey colour, and highlighted entries are shown in orange.

The third tab, 'Websites Visited', shown in Figure 6, displays a bar chart showing the top domain names that appear in the web history. The chart can be configured to show the top 50, top 100 or all visited domain names. The total number of visits to a domain is appended to that domain's bar. A bar chart provides a quick visual indication of general browsing habits, and may help to determine useful filters to apply to other visualisations such as the timeline. The bar chart itself can be restricted using filters. Clicking on a bar displays a pop-up that gives a breakdown of visit numbers by subdomain.

The fourth tab, 'Online Searches', shows a word cloud of all the words entered into search engines in the web history. A word cloud displays a set of words with the font size of each varying depending on how important the word is. In this case, the more a search words occurs, the larger it is displayed, as seen in Figure 7. The set of supported search engines is kept in a configuration file, to which entries can easily be added. Search terms surrounded by quotes are treated as a single word. When words in the word cloud are clicked, a pop-up is displayed showing a list of the full searches the word appeared in, which allows searches for the word to be viewed in context.

Fig. 4. Timeline visualisation. Each blue dot represents a history entry.



A word cloud was chosen because it quickly highlights the most frequently used search terms, as large words attract more attention than smaller words [16]. People tend to scan rather than read word clouds, and so they are more useful for giving quick overall impressions rather than detailed information [17], which can be obtained either by clicking individual words or in conjunction with one of the other visualisations.

The final tab, 'Files' shows all the local files that have accesses recorded in the web history. These only appear in Internet Explorer's `index.dat` files, but are likely to be useful even if Internet Explorer is not used as a browser, because Windows records regular file accesses in `index.dat` [18]. As can be seen from Figure 8, the accessed files are partitioned first by drive letter and then by file type. Under each drive is a pie chart showing the breakdown of the different file types accessed. This is useful to determine what the drive is mainly used for, e.g. many documents and PDFs may suggest a work computer and more images and MP3s may suggest a home computer. For each file type, a tree structure of where the files appeared on the drive is displayed. Each file can be clicked on to show a pop-up giving the dates and times of the accesses.

Tree views are a common way of displaying files in file manager applications, so the presentation of the files here will be familiar. The indentation of folders gives context showing where files are in relation to each other in a way that simply listing the file paths does not.

This visualisation is particularly useful when trying to determine what sorts of files may have been accessed on no longer present USB key drives.

Fig. 5. The timeline graph zoomed on a particular area, with some entries highlighted in orange using filters.

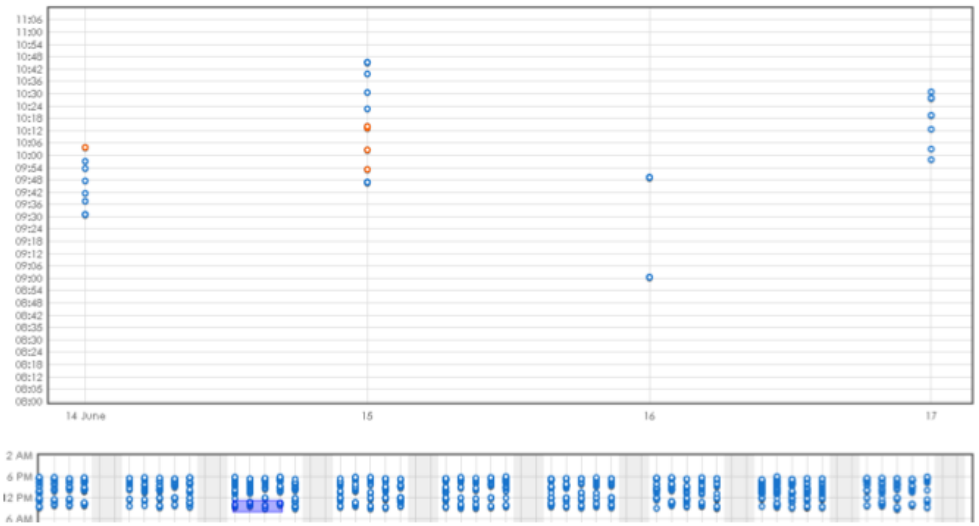
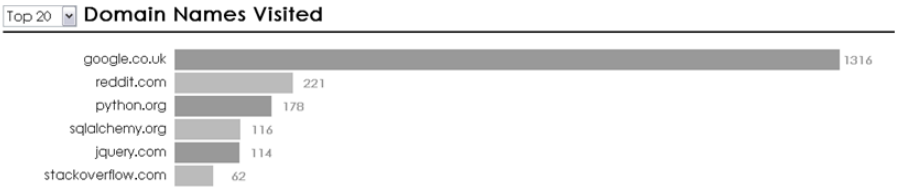


Fig. 6. Bar chart visualisation showing number of domain name visits.



4 User Testing

4.1 Net Analysis

To evaluate the developed visualisations, we performed a small-scale usability study that compared Webscavator with a leading non-visual web browser forensic tool, Net Analysis. Net Analysis "has become the industry standard software for the recovery and analysis of Internet browser artifacts" [19] and is used by many departments including Strathclyde Police, the SCDEA and IBM Incident Response Team. As this tool is widely used, it was chosen as a meaningful non-visual counterpoint to Webscavator.

Net Analysis can import most of the common history files produced by web browsers including Internet Explorer, Firefox, Opera and Safari, as well as some more unusual ones like Yahoo! BT Browser and AOL. Chrome is not yet supported (as of Net Analysis version 1.51). Net Analysis displays its results in a

Fig. 7. Word cloud visualisation showing search terms.

large table with 45 columns, encompassing every field available in all the combined web browsers it supports. Whilst some of the columns will always be populated (e.g. Last Visited and URL), most are only applicable to certain web browsers (e.g. Fav Icon URL).

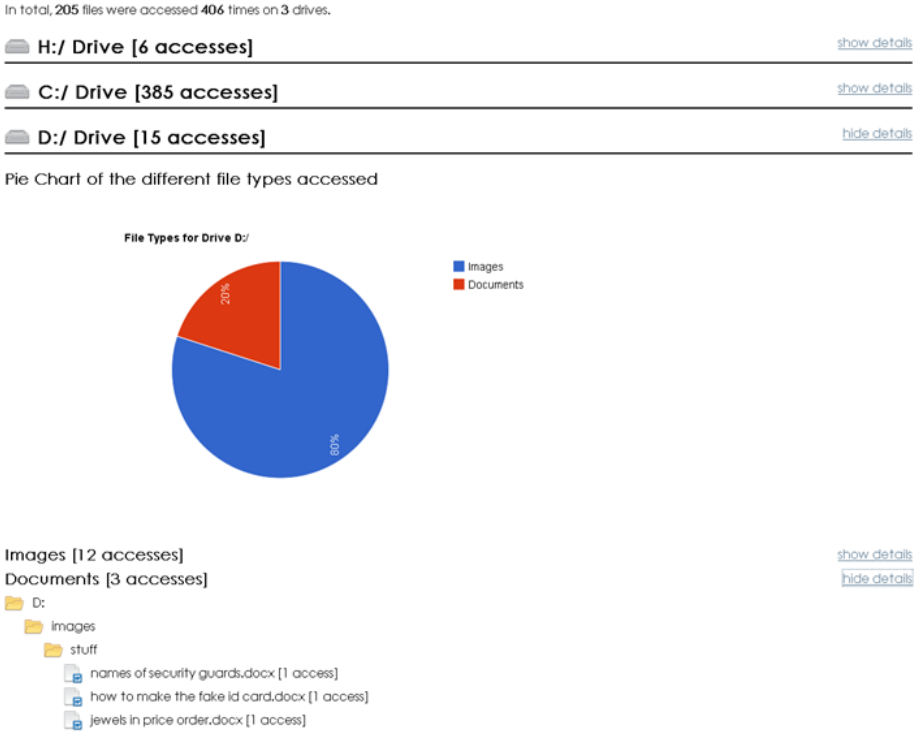
4.2 User Testing Setup

Three digital forensic investigators took part in the user testing. Participants 1 and 2 were digital forensic investigators from a large commercial bank. Participant 1 had some prior knowledge of Net Analysis as he had used it as part of a previous job. Participant 2 had not used Net Analysis before. Both participants used EnCase as their main web browser history analysis tool. Participant 3 was a digital forensic investigator from the Scottish Crime and Drugs Enforcement Agency and had used Net Analysis extensively.

Two scenarios were used for testing, each with three months of invented web history. Scenario A was based on commercial forensic investigations:

> Jane is employed by FatCats Inc and her manager was concerned that her standard of work has gotten increasingly worse and she has missed a lot of deadlines. Since he has spoken to her, her work levels have returned to normal. Even so, he has asked you, FatCat's digital forensic investigator, to look at her recent browser history to see if her decreased productivity was due to her spending her time browsing the internet instead of working. Jane works in IT as a website & python developer. Her regular work hours are 9 - 5.30pm.

Fig. 8. Tree structure and pie chart visualisations for local file accesses.



This scenario emphasised uncovering patterns of usage, comparing web history for different days and discovering the types of websites visited.

Scenario B was based on police forensic investigations:

> On 5th June, a high-end jewellery store in Edinburgh called Diamonte Plaza was robbed. Three thieves nearly got away with over half a million pounds of jewels. Although not present at the crime, John Doe is suspected to have been the mastermind of the operation after one of the thieves confessed after being caught. The computer of John Doe has been seized and the police want to know if John was involved in any way with the robbery.

For each scenario, each participant was asked to answer questions, first using one tool and then the other. Which tool was used first was randomised. To prevent the participants from using their answers discovered using the first program with the second, two seperate web history datasets were developed for each scenario. Which dataset was used with which program was also random. The participants were timed when answering the questions. They were also asked

to mark their answers with a confidence rating from 1 to 5, with 1 indicating a complete guess and 5 indicating total certainty.

As not much quantitative data could be obtained, more emphasis was put on retrieving qualitative data. An unstructured interview took place after the testing. This included a post-task walkthrough during which the participants highlighted any areas of the tools they thought were particularly easy or difficult to use. The 'confidence' rating helped to form questions in the interview.

Finally, the participants were given a survey based on Brooke's System Usability Scale, a mature and extensively used usability questionnaire [20]. This consisted of 20 statements comparing Webscavator to Net Analysis using a rating scale, concentrating on the usability of the programs.

4.3 User Testing Results

In general, participants took less time and made fewer errors when using Webscavator. Confidence was also higher. In Scenario B the participants were 48-55% faster at answering the question set when using Webscavator. A summary of the results for each scenario can be found in Table 1.

Table 1. Summary of the user study results for each scenario.

		Scenario A		Scenario B	
		Net Analysis	Webscavator	Net Analysis	Webscavator
User 1	Correct Answer	67%	100%	33%	100%
	Confidence	37%	77%	64%	78%
	Time	18:54	17:01	17:57	09:50
User 2	Correct Answer	67%	83%	78%	78%
	Confidence	73%	83%	82%	78%
	Time	19:35	10:13	17:12	08:17
User 3	Correct Answer	83%	83%	78%	100%
	Confidence	57%	63%	78%	78%
	Time	14:16	08:42	15:25	07:47

The main difficulties in Net Analysis compared to Webscavator were:

- **Pattern recognition**—especially spotting gaps or densely packed time periods. Webscavator provided many ways to spot patterns, particularly the heat-map and the timeline. Webscavator performed better and faster than Net Analysis for all pattern recognition questions.
- **Summarising data**—counting even small sets of data gave wrong answers. Webscavator gave more accurate and confident results for summarising and collating data.
- **Interpreting search engine queries**—Net Analysis provides no extraction of search terms, meaning the user has to mentally separate out the terms. Webscavator provided a word cloud of search terms, which gave more accurate and faster results.

- **Information overload**—there were a few mistakes in Net Analysis due to participants simply not spotting relevant information. Webscavator's filters are more fine grained and allow for exclusion or highlighting of information. Zooming in also gets rid of irrelevant noise. Information is only visible when the user requires it, i.e. by hovering or clicking, whereas this information is generally always shown in Net Analysis. Participants performed slightly better and had greater confidence with Webscavator on questions requiring timeline usage.

The main difficulty in Webscavator was accurately measuring time periods. This is mainly due to the participants not using filters to highlight the relevant data. Filters were not explained during the Webscavator demo, and so this may explain why two out of the three participants did not try and use them.

To summarise, the visualisations seemed to provide cognitive and perceptive benefits compared with the tabular format of Net Analysis.

4.4 Usability Questionnaire Results

After the user testing was completed, the participants were asked to answer the questions from Brooke's *System Usability Scale* (SUS) for both Net Analysis and Webscavator. There were 10 questions, each with 5 ratings ranging from *strongly disagree* to *strongly agree*. The SUS scale can be converted into a number between 0 to 100 to give an overall sense of the usability of a program [20]. Kortum and Bangor developed an adjective scale based on the SUS to give a better understanding on what the SUS scale means [21]. The results are in Table 2 and Figure 9.

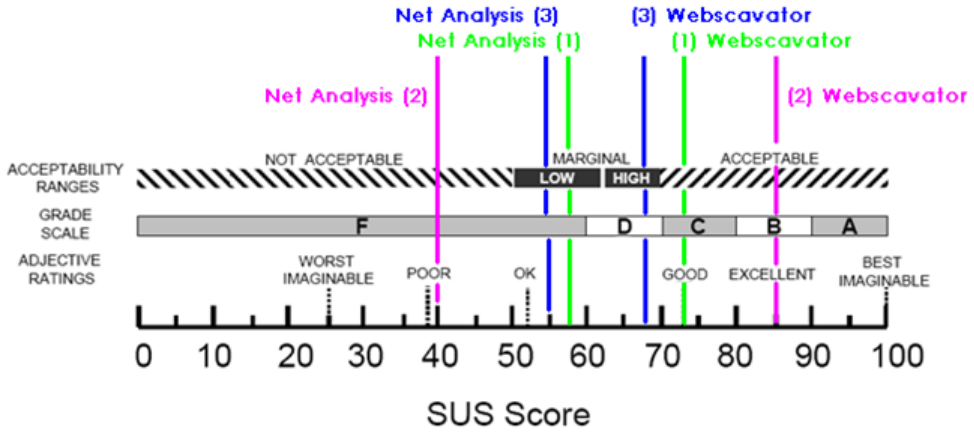
Table 2. SUS and adjective scale for Net Analysis and Webscavator

	Net Analysis		Webscavator	
User 1	57.5	Good (Low)	72.5	Good/Excellent
User 2	40	OK (Not Acceptable)	85	Excellent/Best Imaginable
User 3	55	Good (Low)	67.5	Good (High)

As predicted from the user testing results, Net Analysis scored lower in usability.

A final set of ten questions evaluated the functionality of Webscavator. The questions were designed to determine whether each visualisation was considered useful and whether it would help with real investigations. The overall results were very positive. One of the important aims of Webscavator was to make sure its visualisations were easy to understand by non-technical or non-forensically trained people. Question 9, which asked this, scored a unanimous *agree*. One participant concluded that Webscavator would be invaluable for explaining results to police officers and lawyers, and would even make a good learning tool for computer forensics students.

Fig. 9. Where the results appear on the SUS & adjective scale. Image modified from [21].



5 Conclusions

Investigators face several problems when using the non-visual tools that are currently used to analyse web history data. These include information overload, difficulty spotting correlations and patterns, and difficulty in obtaining a summary overview of data. Visualisation is particularly apt for solving these.

In this paper we introduced Webscavator, a tool that provides several visualisations for examining web history data. User testing results showed that Webscavator's visualisations perform well when compared to Net Analysis, particularly for answering questions that involve spotting patterns or collating and summarising data. Furthermore, a usability questionnaire showed that Webscavator would be a useful tool in digital forensics investigations. All three investigators that took part in the user study expressed a wish to use Webscavator to help with their future work.

These results show that visualisations should be taken seriously by the producers of forensic software. Visualisations can reduce the time it takes to analyse data and reduce the likelihood of errors. There are hundreds of tools available that perform many different forensics functions, but so far not many tools have utilised visualisation. Given that well-chosen visualisations can help significantly in interpreting the large volumes of data produced by many forensic tools, it is likely that the most effective future tools will include visualisations.

6 Further Work

Several improvements were discussed during the interviews following on from user testing. In particular:

1. *Allow filters to be viewed, edited and deleted.* Currently filters can only be added and it is not possible to see the query behind the filter.
2. *Add reporting.* Currently Webscavator is just a visualisation tool, however to be used in practice it needs to be able to generate reports and images. Summaries in PDF format and screenshots as JPEGs would be helpful to investigators.
3. *More integrated visualisations.* Following on from usability questionnaire results, more could be done to integrate the different visualisations. For example clicking on a search term in the word cloud or a domain name in the bar chart could automatically highlight relevant entries in the timeline.
4. *Fuzzy and periodic searching.* The ability to perform a fuzzy search (matches values that are homophones and spelt incorrectly) and a periodical search (matches values that are at regular intervals apart) would be immensely useful. Many people misspell search terms (especially since the likes of Google auto correct spellings), and by filtering on an exact word some searches with spelling errors may be missed. Many employees in a company subscribe to regular, automatic news tickers or RSS feeds, and it would be useful to locate these and remove them from the results.

References

1. AFP. (2008, June 18). Briton Googled 'how to kill' days before murders: court. Retrieved February 14, 2010, from http://afp.google.com/article/ALeqM5hNX7099fMvF7_qHCpy1Bz4VhtR6A
2. Marcella, A. J., & Menendez, D. (2007). Cyber forensics: a field manual for collecting, examining and preserving evidence of computer crimes. CRC Press.
3. Lowman, Sarah (2010). Web History Visualisation for Forensic Investigations. Master's Thesis for Department of Computer & Information Sciences, Uni. Strathclyde.
4. North, C., Stasko, J. T., Fekete, J.-D., & van Wijk, J. J. (2008). The Value of Information Visualisation. Information Visualization: Human-Centered Issues and Perspectives.
5. Larkin, J., & Simon, H. (1987). Why a diagram is (sometimes) worth 10,000 words. Cognitive Science, Volume 11, pp. 65-99.
6. Hendee, W. R., & Wells, P. N. (1997). The perception of visual information. Springer
7. Teerlink, S., & Erbacher, R. (2006). Improving the Computer Forensic Analysis Process through Visualization. Communications of the ACM, Volume 49. Issue 2, pp. 71-75.
8. Shneiderman, B. (1996). The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. IEEE Symposium on Visual Languages, pp. 336-343.
9. Marconini, G. (1997). Information Seeking In Electronic Environments. Cambridge University Press: Issue 9 of Cambridge series on human-computer interaction.
10. Olsson, J., & Boldt, M. (2009). Computer forensic timeline visualization tool. Digital Investigation, Volume 6, pp. S78-S87.
11. Teerlink, S. (2004). A Graphical Representation of File Statistics for Computer Science, Master's Thesis for Computer Science Dept., University of Utah.
12. Stamps, A. S., Franck, J., Carver, J., Jankun-Kelly, T., Wilson, D., & Swan, J. E. (2009). A Visual Analytic Framework for Exploring Relationships in Textual Contents of Digital Forensics Evidence. Proceedings of Workshop on Visualization for Cyber Security, pp. 39-44.

13. Krasser, S., Conti, G., Grizzard, J., Gribschaw, J., & Owen, H. (2005). Real-time and forensic network data analysis using animated and coordinated visualization. Proceedings from the Sixth Annual IEEE Systems, Man and Cybernetics (SMC) Information Assurance Workshop, pp. 42-49.
14. Takada, T., & Koike, H. (2002a). Tudumi: information visualization system for monitoring and auditing computer logs. Proceedings Sixth International Conference on Information Visualisation, pp. 570-576.
15. Wilkinson, L., & Friendly, M. (2009). The history of the cluster heat map. The American Statistician.
16. Lohmann, S., Ziegler, J., & Tetzlaff, L. (2009). Comparison of Tag Cloud Layouts: Task-Related Performance and Visual Exploration. Lecture Notes in Computer Science, Volume 5726.
17. Hassan-Montero, Y., & Herrero-Solana, V. (2006). Improving Tag-Clouds as Visual Information Retrieval Interfaces. International Conference on Multidisciplinary Information Sciences and Technologies.
18. Schatz B., Mohay G. & Clark A.. (2006). A correlation method for establishing provenance of timestamps in digital evidence. Digital Investigation, 3 (Supplement 1). S98-S107
19. DigitalDetective. (2010). Net Analysis. Retrieved June 15th, 2010, from http://www.digital-detective.co.uk/netanalysis.asp
20. Brooke, J. (1996). SUS: a "quick and dirty" usability scale. Usability Evaluation in Industry.
21. Kortum, P., & Bangor, A. (2009). Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. Journal of Usability Studies Vol. 4, Issue 3 , pp. 114-123.